基于全局更新规则蚁群优化的决策实体配置问题求解方法 *

陈冠宇[1]，孙　鹏[1, 2a]，廖梦琛[3]，张杰勇[1]，武君胜[2b]

(1. 空军工程大学 信息与导航学院，西安 710077; 2. 西北工业大学 a. 计算机学院; b. 软件与微电子学院，西安 710072; 3. 中国人民解放军 95445 部队，云南 大理 672100)

摘　要：为解决战场 C2 组织中平台资源的聚类问题，首先定义了决策实体配置过程中各个要素，然后以面向作战任务的形式从外部和内部两个方面定义了决策实体的工作负载，以工作负载的均方根（root mean square，RMS）为目标函数建立决策实体配置模型，将平台调度方案作为输入信息，采用基于全局更新规则的蚁群算法对问题模型进行求解，生成符合作战需求的决策实体配置方案。最后通过仿真算例对所提方法进行验证和分析，实验结果表明，该方法能够得到较好的平台聚类方案，在聚类结果和收敛速度两方面上与传统方法相比均具有优越性。

关键词：决策实体配置；面向作战任务；均方根；蚁群算法

中图分类号：TP391.9　　doi: 10.3969/j.issn.1001-3695.2018.03.0205

Solving method for decision-makers configuration problem based on global update rule of ant colony algorithm

Chen Guanyu[1], Sun Peng[1, 2a], Liao Mengchen[3], Zhang Jieyong[1], Wu Junsheng[2b],

(1. *College of Information & Navigation, Air Force Engineering University, Xi'an* 710077, *China*; 2. *a. School of Computer Science, b. School of Software Micro-electronics, Northwestern Polytechnical University, Xi'an* 710072, *China*; 3. *Unit* 95445 *of People's Liberation Army, Dali Yunnan* 672100, *China*)

Abstract: To solve the problem of clustering of platform resources in C2 organization, firstly, each element in process of decision-making entity configuration is defined, then the work load are defined by two aspects as inside workload and outside workload. The root mean square (RMS) of work load are established as the objective function of configuration model, and platform scheduling scheme is taken as the input information, global updating rule based ant colony algorithm is applied to solve this problem. Finally, the method are verified and analyzed by simulation example, results show that a greater clustering scheme are obtained by this method, and it has advantages over traditional methods in terms of clustering results and convergence speed.

Key words: decision-makers configuration; combat-task-oriented; root mean square; ant colony algorithm

0　引言

随着信息化理论对军事战争的深入影响，如何建立战场环境下平台资源、作战任务、决策实体三者之间的关系，是指挥控制（command and control, C2）组织设计中的关键问题错误!未找到引用源。通过设计作战平台与决策实体之间的指挥控制关系，使不同的平台在相应的决策实体控制下能更好更快地完成该平台所分配到的作战任务，提高任务执行效果和执行质量，就是平台资源的聚类问题，也称为 C2 组织设计中的决策实体配置问题错误!未找到引用源。错误!未找到引用源。。

如何科学合理的对工作负载进行描述和测度，并建立相应的数学模型实现对平台的聚类问题，是 C2 组织决策实体配置过程中主要实现的内容错误!未找到引用源。。文献**错误!未找到引用源。**以最小化决策实体的最大工作负载为目标函数建立了数学模型，从平台、任务数量的角度衡量工作负载；文献**错误!未找到引用源。**以作战任务的执行时间对工作负载进行测度，建立了以工作负载的最小 *RMS* 值为目标函数的数学模型；文献**错误!未找到引用源。**中定义了指控复杂度的概念，从不同平台对作

收稿日期：2018-03-26；**修回日期：**2018-05-10　　**基金项目：**国家自然科学基金资助项目（61573017，61773396）；3 装备预研国防科技重点实验室基金（61421010201）4 陕西省自然科学基金（2017JM6062）

作者简介：陈冠宇（1994-），男，河南郑州人，硕士研究生，主要研究方向为指控组织建模（chenPTN@163.com）；孙鹏（1972-），男，河北安平人，教授，博士研究生，主要研究方向为指控组织建模、体系建模；廖梦琛（1993-），男，云南昆明人，硕士，主要研究方向为指控组织建模、任务规划；张杰勇（1983-），男，浙江湖州人，讲师，博士，主要研究方向为指控组织建模；武君胜（1962-），男，陕西西安人，教授，博士，主要研究方向为指挥信息系统、软件工程、计算机可视化．

战任务的协作情况建立最小化工作负载的问题模型；文献**错误!未找到引用源。**针对传统层次聚类方法采用的贪心策略容易陷入局部最优的现象，提出备选解策略用于选出更加满足整体需求的最优解；针对多目标的优化问题，文献**错误!未找到引用源。**提出对多目标问题模型生成一系列前沿解，并通过一定的筛选方法得到符合要求的平台聚类结果。

本文在以上文献研究内容的基础上，进一步讨论针对作战任务外部工作负载和内部工作负载结合描述决策实体工作负载测度方法，建立以决策实体工作负载 *RMS* 值为目标函数的决策实体配置问题模型，利用基于全局更新规则的蚁群算法对问题模型进行求解，最后通过仿真算例对算法最佳参数设定进行讨论，并对算法的适用性与优越性进行验证与分析，实验结果表明选择合适的算法参数能够得到科学合理的决策实体配置方案，并相较于传统蚁群算法、自适应蚁群算法，该算法具有较快的收敛速度且能得到较好的聚类结果。

1 问题描述及建模

1.1 变量定义

战场环境中的所有实体主要被分为三类，分别为决策实体（DM_m）、平台（P_j）以及作战任务（T_i）。平台聚类问题，即在任务与平台之间的分配关系产生的前提下，实现平台的聚类并将其分配给不同决策实体进行指挥控制的过程。针对本文问题，首先对相关变量作如下定义：

a）平台与作战任务的匹配关系用矩阵 $N=\left(\omega_{ij}\right)_{NT\times NP}(i=1,2,\ldots,NT;j=1,2,\ldots,NP)$ 表示。若 ω_{ij}=1，则表示任务 T_i 需要平台 P_j 执行，若 ω_{ij}=0，则表示任务 T_i 不需要平台 P_j 执行。该矩阵为模型输入已知信息。

b）平台与决策实体的匹配关系用矩阵 $M=\left(m_{nj}\right)_{ND\times NP}(n=1,2,\ldots,ND;j=1,2,\ldots,NP)$ 表示。若 $m_{nj}=1$ 则表示决策实体 DM_m 指挥控制平台 P_j，若 $m_{nj}=0$ 则表示决策实体 DM_m 不对平台 P_j 进行指挥控制。此矩阵即为模型最终输出信息。

c）任务与决策实体的关系用矩阵 x_{ni}=1 $X=(x_{ni})_{ND\times NT}(n=1,2,\ldots,ND;i=1,2,\ldots,NT)$ 表示：

$$x_{ni}=\begin{cases}1,&\text{表示决策实体}DMn\text{执行任务}Ti\\0,&\text{表示决策实体}DMn\text{不执行任务}Ti\end{cases}\qquad(1)$$

x_{ni} 可由 ω_{ij} 和 m_{nj} 的关系推导得出，即若对于平台 P_j，其中 ω_{ij}=1 且 $m_{nj}=1$，则。

d）决策实体之间的协作关系用向量 y_{mni} 表示：

$$y_{mni}=\begin{cases}1,&\text{表示}DM_m\text{和}DM_n\text{在任务}T_i\text{上存在协作}\\0,&\text{表示}DM_m\text{和}DM_n\text{在任务}T_i\text{上不存在协作}\end{cases}\qquad(2)$$

因此，由以上描述可知：

$$y_{mni}=x_{mi}\cdot x_{ni}\qquad(3)$$
$$(m,n=1,2,\ldots,ND,m\neq n;i=1,2,\ldots,NT)$$

e）在执行作战任务中，定义决策实体集合 $DM=\left\{DM_1,DM_2,\ldots,DM_{ND}\right\}$，其中，$ND$ 为决策实体数量；定义所有的作战平台集合 $S^P=\{P_1,P_2,\ldots,P_{NP}\}$，$NP$ 表示作战平台数量；定义所有的作战任务集合 $S^T=\{T_1,T_2,\ldots,T_{NT}\}$，$NT$ 表示所有任务数量。

1.2 约束条件分析

兵力组织的结构可分为决策层、任务层和平台层，这三层中。

a）决策实体和作战平台是一个一对多的关系。一个决策实体可以指挥控制多个作战平台，而一个作战平台只能隶属于一个决策实体。即

$$\sum_{n=1}^{ND}m_{nj}=1\ (\ j=1,2,\ldots,NP)\qquad(4)$$

b）由变量描述可知，当且仅当 $x_{mi}=1$ 且 $x_{ni}=1$ 时，$y_{mni}=1$。因此存在关系

$$y_{mni}\leq x_{mi}\qquad(5)$$
$$(m=1,2,\ldots,ND;n=1,2,\ldots,ND,m\neq n)$$

同理可得

$$y_{mni}\leq x_{ni}\qquad(6)$$
$$(m=1,2,\ldots,ND;n=1,2,\ldots,ND,m\neq n)$$

因为 $x_{ni}=\omega_{ij}\times m_{mj}$，所以

$$y_{mni}\leq\omega_{ij}\cdot m_{mj}\qquad(7)$$

$(m,n=1,2,\ldots,ND,m\neq n;i=1,2,\ldots,NT;j=1,2,\ldots,NP)$ 同理可得：

$$y_{mni}\leq\omega_{ij}\cdot m_{nj}\qquad(8)$$

c）每一个决策实体必须参与指挥控制平台执行该平台相应分配到的作战任务，即每个决策实体指挥控制的平台数必须至少为 1 个：

$$\sum_{j=1}^{NP}m_{nj}\geq1\quad(n=1,2,\ldots,ND)\qquad(9)$$

1.3 问题模型建立

1.3.1 工作负载的定义

工作负载表明了每个决策实体所指挥控制的平台执行任务的情况。考虑到作战任务的完成时间有差异，并且对不同的决策实体带来的工作负载量不同，本文选择以任务执行时间作为工作负载测度**错误!未找到引用源。**，并从内部负载和外部负载两个方面进行描述。

1）内部负载 $I(n)$ 定义及测度方法

决策实体内部负载 $I(n)$ 表示决策实体指挥控制隶属于自己的作战平台完成作战任务所产生的负载。$I(n)$ 数值上等于决策实体控制的所有平台的执行作战任务的时间之和：

$$I(n)=\sum_{j=1}^{NP}m_{nj}\cdot S(j)\qquad(10)$$

其中：$S(j)$ 表示平台 P_j 执行任务的工作负载，定义为作战平台执行任务的时间之和：

$$S(j)=\sum_{i=1}^{NT}\omega_{ij}\cdot t_i \tag{11}$$

其中，t_i 表示任务 T_i 的执行时间。

因此，综上式（10）（11）两式可知：

$$I(n)=\sum_{j=1}^{NP}\left(m_{nj}\cdot\sum_{t=1}^{NT}\omega_{ij}\cdot t_i\right) \tag{12}$$

2）外部负载 $E(n)$ 定义及测度方法

决策实体外部负载表示两个不同决策实体协作完成作战任务而产生的工作负载。$E(n)$ 数值上等于决策实体 DM_n 与其他所有决策实体控制的平台协作执行任务的工作负载之和。类比决策实体内部负载的定义方法将 $E(n)$ 定义为

$$E(n)=\sum_{m=1,m\neq n}^{ND}\sum_{i=1}^{NT}y_{nmi}\cdot t_i \tag{13}$$

3）决策实体 DM_n 总工作负载 $CW(n)$ 测度方法

综合式（12）（13），决策实体总工作负载 $CW(n)$ 的定义为

$$CW(n)=a\cdot I(n)+b\cdot E(n) \tag{14}$$

其中：权重系数 a 和 b 分别为内部负载和外部负载对总工作负载的影响，$a+b=1$。

1.3.2 目标函数构造

由于工作负载从内部负载和外部负载两个方面进行描述，不同的聚类方案中决策实体指挥和控制作战平台完成任务的工作负载的不同将对配置方案产生不同的结果。对于某一个给定的平台调度方案，无论聚类结果如何变化，所有决策实体的内部负载之和是保持不变的，但不同的聚类方案决策实体的外部负载之和却不同。因此可以通过选择合适的聚类方案使得所有决策实体的外部负载之和最小。本文选择用每一个决策实体平均工作负载最小、同时决策实体之间负载差异最小作为聚类模型的目标函数，其中，决策实体的平均工作负载表示为

$$\mu=\frac{1}{D}\sum_{m=1}^{D}CW(m) \tag{15}$$

决策实体之间的差异由方差表示，即

$$\sigma^2=\frac{1}{D}\sum_{m=1}^{D}[CW(m)-\mu]^2 \tag{16}$$

因此，用均方根 RMS 值对决策实体的工作负载进行综合描述，即

$$\begin{aligned}
\text{RMS}&=\sqrt{\frac{1}{D}\sum_{m=1}^{D}CW(m)^2}=\sqrt{\frac{1}{D}\sum_{m=1}^{D}CW(m)^2+2\mu^2-2\mu^2}\\
&=\sqrt{\mu^2+\frac{1}{D}\sum_{m=1}^{D}CW(m)^2+\frac{\mu^2}{D}\cdot D-2\mu\cdot\frac{1}{D}\sum_{m=1}^{D}CW(m)}\\
&=\sqrt{\mu^2+\frac{1}{D}\sum_{m=1}^{D}CW(m)^2+\sum_{m=1}^{D}\frac{\mu^2}{D}-2\mu\cdot\frac{1}{D}\sum_{m=1}^{D}CW(m)}\\
&=\sqrt{\mu^2+\frac{1}{D}\sum_{m=1}^{D}\left(CW(m)^2+\mu^2-2\mu\cdot CW(m)\right)}\\
&=\sqrt{\mu^2+\frac{1}{D}\sum_{m=1}^{D}[CW(m)-\mu]^2}\\
&=\sqrt{\mu^2+\sigma^2}
\end{aligned} \tag{17}$$

1.3.3 数学模型建立

由式(14)可知，决策实体的工作负载将通过均值和方差两方面进行描述，因此，在对问题模型的约束条件分析及目标函数构造的基础上，本文建立的决策实体配置问题数学模型表示为

$$\min\ \text{RMS}=\sqrt{\frac{1}{D}\sum_{m=1}^{D}CW(m)^2} \tag{18}$$

$$\begin{aligned}
s.t.\ &\sum_{n=1}^{ND}m_{nj}=1,\ \ j=1,2,\ldots,NP\\
&\sum_{n=1}^{NP}m_{nj}\geq 1,\ \ i=1,2,\ldots,NT;\\
&y_{mni}\leq\omega_{ij}\cdot m_{mj},\ \ m=1,2,\ldots,ND;\ \ n=1,2,\ldots,ND\\
&y_{mni}\leq\omega_{ij}\cdot m_{nj},\ \ m=1,2,\ldots,ND;\ \ n=1,2,\ldots,ND
\end{aligned}$$

2　基于全局更新规则蚁群算法的决策实体配置问题求解

决策实体配置就是实现平台聚类的过程，即把不同的平台分配给一定数量的决策实体，由决策实体控制各个平台执行既定的作战任务。在平台聚类过程中，要在满足问题约束条件的同时得到符合问题模型目标函数的最优解，因此平台聚类方案体现了所设计的平台与决策实体之间配置关系的合理性。

决策实体的配置问题本质上是组合优化问题，可以采用智能算法对问题进行求解。蚁群算法(ant colony optimization, ACO)，作为一种具有搜索能力强、收敛速度快等特点的智能算法，已经被广泛应用于求解组合优化问题。在路径规划[错误!未找到引用源。]、流水车间调度[错误!未找到引用源。]、最优解搜寻[错误!未找到引用源。]、建筑选址[错误!未找到引用源。]和无人机作战目标分配[错误!未找到引用源。]等问题上有着较好的表现，因此蚁群算法可以作为解决平台聚类过程中寻求最优聚类方案的一种方法。

该算法按照启发式思想,使用正反馈和信息素挥发机制，利用蚂蚁在所通过的路径上释放的信息素的诱导作用,通过多次迭代逐步收敛到所求解问题的全局最优解。算法的核心思想可以简要描述为：所有蚂蚁都会在经过的路径上留下信息素，不同蚂蚁行走的路径对应不同的信息素轨迹。由于在最短路径上的蚂蚁会更早的到达目的地，从而增加其在该路径上来往的频率，因此该路径上信息素密度更大。其他蚂蚁在寻径的过程中会选择信息素强的路径进行移动，因此，此路径上的信息素强度会不断叠加。通过这种正反馈机制使蚂蚁最终能够找到蚁穴到食物源的最优路径。

下面将针对决策实体的配置问题，结合文献**错误!未找到引用源。**，对蚁群算法在实现平台聚类过程中算法各个组成要素进行相应的描述。

a）构建图。蚂蚁将在构建矩阵 C 上随机游走，来建立解的随机构建过程。其中，L 完全连接集合 C 中所有元素，C 中包含所有平台和决策实体。一个可行解就表示一个由 ND 个决策实体和 NP 个平台以 (m,j) 的配对形式建立的矩阵，若 (m,j)

等于 1，则表示平台 P_j 分配给决策实体 DM_m 进行指挥控制。分配过程中，一个平台只能分配给一个决策实体，而一个决策实体可以指挥控制多个平台。

b）信息素。信息素 τ_{mj} 表明平台 P_j 到决策实体 DM_m 的期望度，它编码了有关蚂蚁进行路径搜索的过程，由蚂蚁对其进行更新。

c）信息素全局更新。为了避免冗余的信息素造成过量的信息堆积，每只蚂蚁完成对 NP 个平台的聚类后将淡化旧信息，强化新信息。因此，在 $t+n$ 时刻，平台与决策实体之间的配对 (m,j) 将按全局更新规则进行更新：

$$\tau_{mj}(t+n)=(1-\rho)\cdot\tau_{mj}(t)+\Delta\tau_{mj}(t,t+n) \tag{19}$$

其中: ρ 为挥发系数，表示信息素随时间推移的衰减程度；$\Delta\tau_{mj}$ 表示信息素增量。

$$\Delta\tau_{mj}(t,t+n)=\sum_{k=1}^{M}\Delta\tau_{mj}^{k}(t,t+n) \tag{20}$$

$\Delta\tau_{mj}^{k}(t,t+n)$ 表示第 k 只蚂蚁在 $(t,t+n)$ 时刻留在路径 (m,j) 上的信息素量，具体数值根据蚂蚁的表现而定。计算公式为

$$\Delta\tau_{mj}^{k}(t,t+n)=\begin{cases}\dfrac{Q}{RMS_k}, & \text{若第}k\text{只蚂蚁经过路径}(m,j)\\ 0, & \text{其他}\end{cases} \tag{21}$$

其中：Q 表示控制常数，RMS_k 表示每只蚂蚁表示的分配方案的工作负载 RMS 值：

$$RMS=\frac{1}{ND}\sum_{N=1}^{ND}CW(n)^2=\sqrt{\mu^2+\sigma^2} \tag{22}$$

d）结束规则。如果所有平台都已分配完毕，或者算法迭代次数超过某个迭代次数限制，则结束。

综上所述，基于全局更新规则蚁群算法的决策实体配置过程步骤如下：

a) 初始化所有参数。设置蚂蚁数量为 N，最大循环次数为 N_{stop}，初始化每个配对 (m,j) 的信息素 $\tau_{mj}(0)$，令初始时刻 $\Delta\tau_{mj}(0)=0$。N 只蚂蚁随机产生决策实体与平台之间的配置方案 C，对方案的可行性进行验证，剔除不可行的配置方案；

b) 计算不同的决策实体配置方案中决策实体的工作负载 RMS 值；

c) 计算信息素增量 $\Delta\tau_{mj}^{k}(t,t+n)$，求解信息素 $\tau_{mj}(t+n)$，并根据信息素计算下式的转移概率:

$$P_{mj}^{k}=\frac{\tau_{mj}^{\alpha}(t)\eta_{mj}^{\beta}(t)}{\sum_{s\in allowed_k}\tau_{ms}^{\alpha}(t)\eta_{ms}^{\beta}(t)} \tag{23}$$

d) 按照转移概率进行聚类方案选择，记录当前最优分配方案；

e) 判断循环次数是否小于最大循环次数，若小于则保存最优个体，返回执行步骤 2，若大于则输出最后结果。

基于蚁群算法的平台聚类过程如图 1 所示。

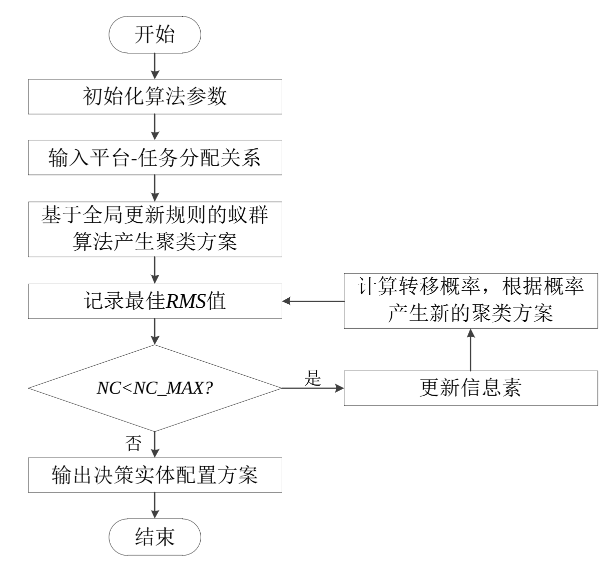


图 1　基于全局更新规则蚁群算法的聚类过程

3　算例分析

本文以一个联合作战作战想定为仿真算例，对本文所提方法进行仿真验证分析。在该案例中，共有 18 个任务、20 个平台，其中平台资源能力与任务资源需求的种类为 8 种，任务属性数据和平台属性数据参见文献**错误!未找到引用源。**。其中，任务-平台间的分配关系用平台调度方案甘特图进行表示，该分配关系为决策实体配置问题的已知输入信息，如图 2 所示。

时间（0–140）	任务
P1	T17, T12
P2	T1, T2
P3	T2
P4	T7, T11, T15, T16
P5	T17, T8, T15
P6	T1
P7	T4, T11, T14
P8	T1, T2
P9	T3, T7, T8, T12
P10	T6, T7, T9, T13, T15, T10, T16
P11	T5, T13
P12	T5, T9, T13
P13	T5, T8, T10
P14	T5, T18, T12
P15	T17, T2, T14
P16	T1, T2, T14
P17	T18, T13
P18	T6, T18, T16
P19	T6, T7, T15
P20	T17, T8, T16

图 2　平台调度方案甘特图

仿真实验 1 在基于全局更新规则的蚁群算法中，由于算法涉及到的参数较多，且不同参数设置的结果将对最终的平台聚类结果产生较大影响，因此首先针对不同算法参数对聚类结果的影响进行比较分析，确定解决该问题的最佳参数设置问题。

1）决策实体数量 *ND* 对聚类结果的影响

算法初始化参数设置如下：蚂蚁数量 $M=25$、平台个数 $NP=20$、任务个数 $NT=18$、最大循环次数 $NC_{max}=50$，信息素残留系数 $Rho=0.8$、信息素总量 $Q=100$、内部负载权重 $a=0.5$、外部负载权重 $b=0.5$、信息启发式因子 $\alpha=0.8$、期望值启发式因子 $\beta=3$。令决策实体数量分别为 $ND=2,3,\ldots,10$，工作负载 *RMS* 值随 *ND* 变化情况如图 3 所示。

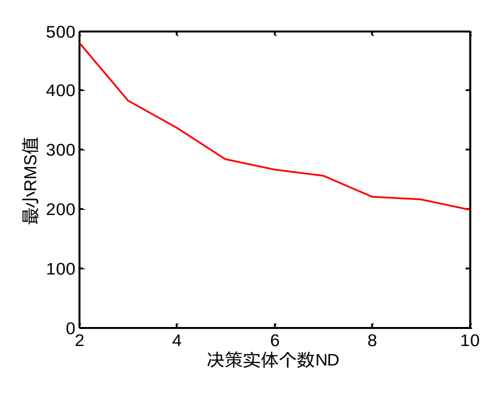


图 3　RMS 值随 ND 个数变化曲线图

由图 3 可知，决策实体数量将影响目标函数值的变化勤快，一般决策实体的个数越少，目标函数 *RMS* 的值就越大，表明当决策实体数量少时，每个决策实体的负载越大。但是并不是决策实体的个数越多，聚类方案就越好。受到作战实际的约束，决策实体个数是有限的，在一次战役任务中，过多决策实体将造成决策实体成本增加。由曲线变化趋势可以看出，当决策实体在 4-6 个之间时，曲线的斜率变化最快，因此决策实体个数应该选择 4-6 个。综合考虑，选择决策实体数量 $ND=5$ 作为后续聚类过程的参数值。

2）信息素残留系数 ρ 对聚类结果的影响

信息素挥发度是指蚂蚁在走过的路径上留下的信息素的挥发程度。在算法中以 $1-\rho$ 为信息素挥发系数，即以 ρ 为信息素残留系数。信息素挥发系数直接影响到蚁群算法的收敛速度和全局搜索能力。由于信息挥发系数的存在，蚂蚁没有走过的路径上的信息素会逐渐挥发直至减少到 0。当信息素挥发系数 $1-\rho$ 过大时，信息素挥发过快，一些蚂蚁未经过路径不会得到选择，降低了算法的全局搜索能力；而当信息素挥发系数 $1-\rho$ 过小时，虽然能提高算法的全局搜索能力，但是算法收敛速度会降低。信息素残留系数对算法性能的影响如表 1 所示

表 1　信息素残留系数对算法性能的影响

残留系数 ρ	最优 *RMS* 值	收敛代数
0.1	288.6357	20
0.3	288.7796	15
0.5	288.2410	11
0.7	275.6084	15
0.8	276.3512	16
0.9	267.0019	21
1	270.2591	19

信息素残留系数 ρ 与算法输出 *RMS* 的之间关系如图 4 所示。

收稿日期：2018-03-26；**修回日期**：2018-05-10　**基金项目**：国家自然科学基金资助项目（61573017，61773396）；3 装备预研国防科技重点实验室基金（61421010201）4 陕西省自然科学基金（2017JM6062）

作者简介：陈冠宇（1994-），男，河南郑州人，硕士研究生，主要研究方向为指控组织建模（chenPTN@163.com）；孙鹏（1972-），男，河北安平人，教授，博士研究生，主要研究方向为指控组织建模、体系建模；廖梦琛（1993-），男，云南昆明人，硕士，主要研究方向为指控组织建模、任务规划；张杰勇（1983-），男，浙江湖州人，讲师，博士，主要研究方向为指控组织建模；武君胜（1962-），男，陕西西安人，教授，博士，主要研究方向为指挥信息系统、软件工程、计算机可视化.

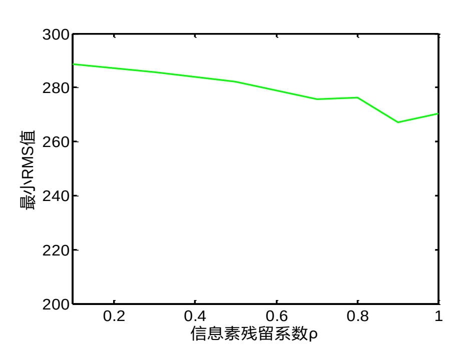


图 4　信息素挥发系数与输出解间的关系

信息素残留系数 ρ 与算法收敛迭代次数之间的关系如图 5 所示。

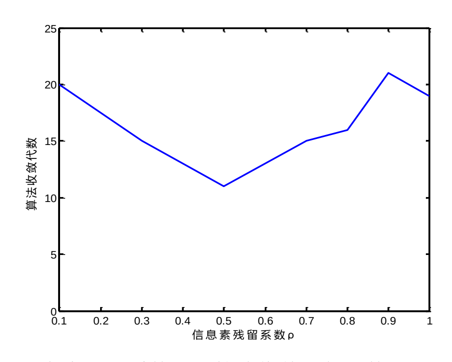


图 5　信息素残留系数 ρ 与算法收敛迭代次数之间的关系

由曲线趋势可以看出，算法在 $\rho = 0.7-0.9$ 间时，输出 RMS 最小。因此本设计综合其收敛速度和全局搜索能力等因素。综合考虑，选择信息素挥发系数 $1-\rho = 0.3$，即信息素残余系数 $\rho = 0.7$ 作为后续聚类过程的参数值。

3）蚂蚁数量 M 对聚类结果的影响

蚁群算法中，蚂蚁的数量的多少直接影响着群体表现出的能力。对于单个蚂蚁而言完成一次分配即为一个有效解，所有蚂蚁完成一次分配可构成一个解集。显然，蚂蚁的数量越多，解集越大，算法的全局搜索能力就越强，但是算法的收敛速度就越慢。反之，蚂蚁的数量越少，解集越小，算法的全局搜索能力就越弱，但是算法的收敛速度就越快。当蚂蚁数量过小时，虽然提高了算法的收敛速度，但是却容易使算法陷入局部最优，全局搜索效能降低，容易使算法过早停滞现象。蚂蚁数量 M 对对聚类结果及算法收敛情况的影响如表 2 所示。

表 2 蚂蚁数量 M 与 RMS 值、算法迭代次数关系表

蚂蚁数量 M	最优 RMS 值	算法收敛代数
20	269.6683	18
25	256.4048	20
30	278.9426	22
35	273.5055	25
40	278.3597	30
50	259.5824	34
80	272.2179	40

由表 2 可知，当蚂蚁数量过多时，收敛速度降低的同时会影响算法输出的聚类结果。由图可以看出选择蚂蚁只数 $M = 25$ 时，输出结果较小、迭代次数较少，因此设置参数 $M = 25$ 作为后续聚类过程的参数值。

4）启发因子 α、β 的选择

信息启发式因子 α 反映蚂蚁在运动过程中所积累的信息量在指导蚁群搜索中的相对重要程度，期望值启发式因子 β 反映蚂蚁在运动过程中启发信息在指导蚁群搜索中的相对重要程度。期望值启发式因子 β 的值越大，蚂蚁在某个局部点上选择局部最短路径的可能性越大，算法收敛速度得以加快，但蚁群在最优路径的搜索过程中随机性减弱，易于陷入局部最优；而信息启发式因子 α 的大小则反映了蚁群在路径搜索中随机性因素作用的强度，其值越大，蚂蚁选择以前走过的路径的可能性越大，搜索的随机性减弱，当值过大也会使蚂蚁的搜索过早陷于局部最优。启发因子 α 和 β 对输出结果及算法收敛性的影响如表 3 所示。

表 3 启发因子 α 和 β 对输出结果及算法收敛性的影响

信息启发因子 α	期望值启发因子 β	最优 RMS 值	算法收敛代数
0.1	0.1	281.5496	80
0.3	0.5	282.9456	40
0.5	1.5	266.1665	25
0.8	3	265.3575	18
1	5	266.4062	12
3	8	272.4683	12
5	10	296.6914	6

由表 3 可知当信息启发因子 $\alpha = 0.5-3$，同时期望值启发因子 $\beta = 3$ 左右时算法的输出结果较为优越，且算法收敛速度较快，因此设置参数 $\alpha = 0.8$、$\beta = 3$ 作为后续聚类过程的参数值。

仿真实验 2　利用实验 1 中的结果，对算法中的参数进行初始化设置：蚂蚁数量 $M = 25$、决策实体数量 $ND = 5$、平台个数 $NP = 20$、任务个数 $NT = 18$、最大循环次数 $NC_{max} = 50$、信息素残留系数 $Rho = 0.8$、信息素总量 $Q = 100$、内部负载权重 $a = 0.5$、外部负载权重 $b = 0.5$、信息启发式因子 $\alpha = 0.8$、期望值启发式因子 $\beta = 3$。利用基于全局更新规则的蚁群算法，生成以图 1 中所示的平台-任务分配关系为输入信息的平台聚类方案，其聚类结果如表 4 所示。

表 4　基于全局更新规则蚁群算法的平台聚类结果

决策实体	平台	DM 工作负载	工作负载 RMS 值	RMS 方差
DM_1	$P_1, P_4, P_5, P_9, P_{20}$	260		
DM_2	$P_2, ,P_3, P_6, P_8, P_{15,} P_{16}$	350		
DM_3	P_7, P_{13}	130	256.2421	80.1873
DM_4	$P_4, P_{11}, P_{12}, P_{14}, P_8$	220		
DM_5	P_{10}, P_{18}, P_{19}	270		

算法收敛曲线如图 6 所示。

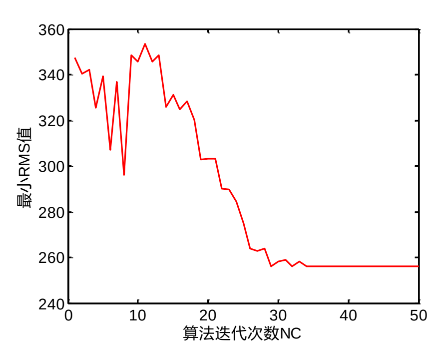


图 6　蚁群算法收敛曲线与迭代次数图

由图 6 及表 4 可知，基于全局更新规则的蚁群算法在解决平台聚类问题时具有可行性。从算法输出结果可以看出，决策实体的工作负载平均值为 246，工作负载 *RMS* 值为 256.2421，决策实体工作负载方差为 80.1873。从算法收敛曲线可以看出，算法在迭代至 33 代的时候收敛。因此，蚁群算法在解决解决平台聚类问题上具有较好的性能，验证了该方法的可行性和适用性。

仿真实验 3 为了进一步验证基于全局更新规则的蚁群算法的优越性，首先将本文所提算法与层次聚类算法、自适应蚁群算法进行对比分析。算法中的关键参数设置为：蚂蚁数量 $M=25$、决策实体数量 $ND=5$、平台个数 $NP=20$、任务个数 $NT=18$、最大循环次数 $NC_{max}=50$、信息素残留系数 $Rho=0.8$、信息素总量 $Q=100$、内部负载权重 $a=0.5$、外部负载权重 $b=0.5$、信息启发式因子 $\alpha=0.8$、期望值启发式因子 $\beta=3$。采用层次聚类法和自适应蚁群算法的输出平台聚类结果如表 5、表 6 所示：

表 5　传统层次聚类法解决平台聚类问题的输出结果

决策实体	平台	DM 工作负载	工作负载 RMS 值
DM_1	$P_1, P_5, P_9, P_{13}, P_{20}$	240	
DM_2	$P_2, ,P_6, P_8,, P_{16}$	280	
DM_3	P_3, P_7, P_{15}	200	258.1859
DM_4	$P_4, P_{10}, P_{18}, P_{19}$	330	
DM_5	$P_{11}, P_{12}, P_{14}, P_{17}$	220	

表 6　自适应蚁群算法解决平台聚类问题的输出结果

决策实体	平台	DM 工作负载	工作负载 RMS 值
DM_1	$P_3, P_7, P_{15}, P_{13,}$	200	
DM_2	$P_1, ,P_4, P_5, P_{9,} P_{20}$	260	
DM_3	$P_{10}, P_{11}, P_{12,} P_{17,} P_{18}, P_{19}$	360	257.8759
DM_4	P_{13}, P_{14}	130	
DM_5	P_2, P_6, P_8, P_{16}	280	

由表 4~6 对比结果可知，采用全局更新规则的蚁群算法得到的最终聚类目标函数值为 RMS=256.2421，层次聚类法聚类结果为 RMS=258.1859，自适应蚁群算法聚类结果为 RMS=257.8759。就聚类结果而言，采用全局更新规则的蚁群算法得到的聚类方案 RMS 值最小，自适应蚁群算法次之，传统层次聚类法最大。因此蚁群算法在解决平台聚类问题上优于传统层次聚类法。而自适应蚁群算法在求解该问题时的效果不如采用全局更新规则的蚁群算法。进一步证实了蚁群算法在解决平台聚类问题上的可行性和优越性。

对该问题分别采用全局更新规则的蚁群算法、自适应蚁群算法、普通蚁群算法和量子遗传算法(quantum genetic algorithm, QGA)进行求解，对算法收敛情况进行进一步比较分析，各算法收敛曲线如图 7 所示。

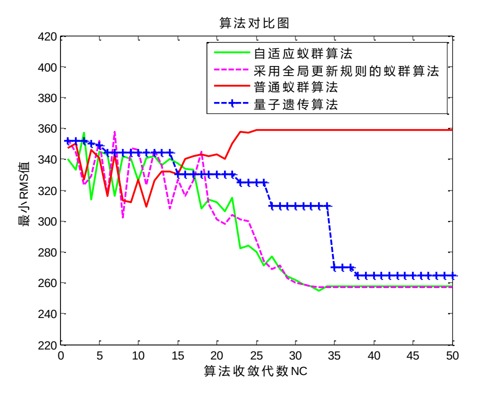


图 7　不同算法收敛代数比较图

由图 7 中各算法收敛曲线趋势可以直观看出，采用全局更新策略的蚁群算法收敛效果与采用自适应蚁群算法的收敛效果类似，全局更新策略的蚁群算法的收敛效果和达到收敛的迭代数均优于量子遗传算法，因此，进一步验证了采用全局更新策略的蚁群算法在求解平台聚类模型时具有优越性。

4　结束语

本文主要解决了 C2 组织中的决策实体配置问题，亦平台聚类问题。首先以面向作战任务的形式从外部和内部两方面定义了决策实体的工作负载，并以工作负载的 *RMS* 值为目标函数建立了决策实体配置问题模型，采用基于全局更新规则的蚁群算法对该问题进行求解，最后通过仿真算例验证了算法的适用性与优越性。实验结果表明，合理的算法参数设置能够得到性能较好的聚类结果，并且该方法与传统层次聚类法能够得到更小的目标函数值，且从收敛速度来看该方法在解决平台聚类问题时优于传统蚁群算法和自适应蚁群算法，能够较好的解决决策实体配置问题。在下一步的工作中，将针对不确定环境下的决策实体配置方案优化调整问题进行研究。

参考文献：

[1] Bui H, Han X, Mandal S, *et al*. Optimization-based decision support algorithm for a team-in-the-loop planning experiment [C]// Proc of IEEE international Conference on Systems, Man, and Cybernetics, 2009: 4685-4689.

[2] 阳东升，张维明，刘忠，等. 指控组织设计方法 [M]. 北京：国防工业出版社, 2010: 161-189.

[3] Han X, Mandal S, Pattipati K R, *et al*. An optimization-based distributed planning algorithm: A blackboard-based collaborative Framework [J]. IEEE Trans on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2014, 44 (6): 673-684.

[4] 孙昱，姚佩阳，张杰勇. C2 组织信息结构效能测度及综合评估 [J]. 系统工程与电子技术，2015, 37 (6): 1313-1318. (Sun Yu, Yao Peiyang, Zhang Jieyong. Measurement and comprehensive evaluation of C2 organizational information structure efficiency [J]. Systems Engineering and Electronics, 2015, 37 (6): 1313-1318.)

[5] David S, Kevin S. On optimizing command and control structures [C]// Proc. of the 16th International Command and Control Research Symposium. 2011: 1-10.

[6] Levchuk G M, Levchuk Y N, Luo J, *et al.* Normative design of organizations—Part Ⅰ: mission planning [J]. IEEE Trans on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2002, 32 (3): 346-359.

[7] Levchuk G M, Levchuk Y N, Luo J, *et al.* Normative design of organizations—Part Ⅱ: organizational structure [J]. IEEE Trans. on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2002, 32 (3): 360-375.

[8] 张杰勇，姚佩阳. C2 组织决策实体配置问题建模与求解方法 [J]. 系统工程与电子技术，2012, 34 (4): 737-742. (Zhang Jieyong, Yao Peiyang. Model and solving method for collocating problem of decision-makers in C2 organization [J]. Systems Engineering and Electronics, 2012, 34 (4): 737-742.)

[9] 吴瑞杰，孙鹏，孙昱，等. 自适应量子遗传算法在指挥控制结构设计中的应用 [J]. 计算机应用与研究, 2016, 33 (7): 113-119. (Wu Ruijie, Sun Peng, Sun Yu, *et al.* Self-adaption quantum genetic algorithm used in design of command and control structure [J]. Application Research of Computers, 2016, 33 (7): 113-119.)

[10] 周翔翔，姚佩阳，尹忠海. 指挥决策实体层次结构设计方法 [J]. 电光与控制，2012, 19 (2): 68-72. (Zhou Xiangxiang, Yao Peiyang, Yin Zhonghai. A method for design of command decision-making entity’s hiberarchy architecture [J]. Electronics Optics & Control, 2012, 19 (2): 68-72.)

[11] 孙昱，姚佩阳，吴吉祥等. 兵力组织扁平化指挥控制结构设计方法 [J]. 系统工程与电子技术，2016, 38 (8): 1833-1839. (Sun Yu, Yao Peiyang, Wu Jixiang, *et al.* Design method of flattening command and control structure of army organization [J]. Systems Engineering and Electronics, 2016, 38 (8): 1833-1839.)

[12] Naz-afarin M, Vahidreza G. Integration of efficient multi-objective ant-colony and a heuristic method to solve a novel multi-objective mixed load school bus routing model [J]. Applied Soft Computing, 2018, 68: 92-109.

[13] 王圣尧，王凌，许烨，等. 求解混合流水车间调度问题的分布估计算法 [J]. 自动化学报, 2012, 38 (3): 437-443. (Wang Shengyao, Wang Ling, Xu Ye, *et al.* An estimation of distribution algorithm for solving hybrid flow-shop scheduling problem [J]. Acta Automatica Sinica, 2012, 38 (3): 437-443.)

[14] Soumya B, Abhishek P. Test Suite Minimization in Regression Testing Using Hybrid Approach of ACO and GA [J]. International Journal of Applied Metaheuristic Computing, 2018, 9 (3): 88-104.

[15] Xin Ning, Jingyan Qi, Chunlin Wu, Wenjuan Wang. A tri-objective ant colony optimization based model for planning safe construction site layout [J]. Automation in Construction, 2018, 89: 1-12.

[16] Ziyang Zhen, Dongjing Xing, Chen Gao. Cooperative search-attack mission planning for multi-UAV based on intelligent self-organized algorithm [J]. Aerospace Science and Technology, 2018.

[17] 张邦楚，彭琛，余新荣等. 基于蚁群优化的多弹协同目标分配算法 [J]. 弹箭与制导学报，2012, 32 (4): 69-73. (Zhang Bangchu, Peng Chen, Xu Rongxin, *et al.* Cooperative target assignment algorithm of multiple missiles based on ant colony optimization [J]. Journal of Projectiles, Rockets, Missiles and Guidance, 2012, 32 (4): 69-73.)